# Chương 2.

1. Ràng buộc toàn vẹn là gì?

Tập các quy tắc mà mọi dữ liệu trong CSDL phải tuân theo nhằm đảm bảo tính toàn vẹn của cơ sở dữ liệu.

1. Tại sao các quan hệ phải có ràng buộc toàn vẹn?

Để đảm bảo tính toàn vẹn của một tập dữ liệu

1. Khóa chính (primary key) của một quan hệ Q

Một hoặc nhiều thuộc tính dùng để xác định một thực thể trong tập thực thể

1. Khóa ngoại (foreign key) của một quan hệ Q

Một hay nhiều thuộc tính (khóa chính hay thuộc tính có ràng buộc unique) trong một quan hệ A xuất hiện trong một quan hệ B khác được gọi là khóa ngoại trong quan hệ B

1. Ràng buộc về miền giá trị của 1 thuộc tính (Domain)

Là tập các giá trị mà thuộc tính đó có thể nhận

1. Một thuộc tính có ràng buộc Unique?

Là thuộc tính mà các thực thể có thuộc tính này là riêng biệt

1. Một thuộc tính có ràng buộc not null nghĩa là?

Là thuộc tính đó không được để trống

1. Một thuộc tính có ràng buộc Check (điều kiện) nghĩa là?

Là thuộc tính phải thỏa điều kiện check của rảng buộc đó

1. Câu lệnh khai báo, thêm và xóa các ràng buộc của quan hệ Q.

Khai báo: col datatype Check(condition)

Thêm: Altering table\_name add constraints condition\_name condition(col)

Xóa: Altering table\_name drop constraints condition\_name

1. Cho Q1(A int, B int), Q2(A int, C int). A trong Q2 là khóa ngoại tham chiếu qua A của Q1. Thao tác nào dưới đây không thực hiện được
2. Thêm 1 dòng có A = 1 vào Q2, nhưng trong Q1 chưa có dòng nào có A = 1.
3. Sửa 1 dòng trong Q2 có A = 1 thành A=0, nhưng trong Q1 chưa có dòng nào có A = 0.
4. Cả (1) và (2) đều đúng
5. Cả (1) và (2) đều sai
6. Trigger là gì?

**Trigger trong SQL** chính là một đoạn chương trình con, chúng sẽ vận hành chỉ khi có một sự kiện nào đó xảy ra

1. Ý nghĩa của biến cố, điều kiện và hành động trong trigger

Trigger chỉ được kích hoạt khi có một biến cố (event) nhất định được người lập trình sử dụng xảy ra (thường là insert, delete, update)

Khi được kích hoạt, trigger sẽ kiểm tra điều kiện.

Điều kiện nếu thỏa, hành động sẽ được thực hiện bởi hệ quản trị cơ sở dữ liêu.

1. Ý nghĩa của For và After trong câu lệnh tạo trigger

After được kích hoạt khi tất cả các hoạt động INSERT, UPDATE, DELETE được thực hiện, các referential cascade cần được thực hiện trước khi trigger được thực hiện. After được định dạng sẵn còn For là từ khóa duy nhất được specified. After không được định nghĩa trong Views.

1. Ý nghĩa của câu lệnh dưới

Create trigger AA on Q after delete as declare @new int, @old int select @old=ol.B from deleted ol if (@old >2) print ('Notify Customer Relations');

Tạo ra trigger tên AA trong bảng Q, trigger được thực hiện sau khi thực hiện lệnh xóa với các biến ol.B; tạo hai biến new, old = ol.b (vừa được xóa); nếu biến old lớn hơn 2 thì in ra Notify Customer Relations

1. Instead of trigger là gì?

Chỉ định rằng trigger DML được thực thi thay vì câu lệnh trigger SQL, do đó, ghi đè các hành động của câu lệnh trigger. INSTEAD OF không thể được chỉ định cho các lệnh trigger DDL hoặc trigger logon. Tối đa, một trigger INSTEAD OF cho mỗi câu lệnh INSERT, UPDATE hoặc DELETE có thể được xác định trên một bảng hoặc dạng xem. INSTEAD OF trình kích hoạt không được phép trên các dạng xem cập nhật có sử dụng With check option.

1. Ý nghĩa của câu lệnh dưới

Create trigger AA on Q after delete as declare @old int select @old=ol.B from deleted ol if (@old >2) Begin print ('Notify Customer Relations'); Rollback; end

Tạo ra trigger AA trên bảng Q: sau khi xóa biến ol, khai báo biến old = ol.B (vừa được xóa). Nếu old lớn hơn 2, thì sau đó, in ra “Notify Customer Relations” và khôi phục lại dữ liệu vừa xóa.

1. Cho

Create trigger AA on Q after delete as declare @old int select @old=ol.B from deleted ol if (@old >2) Begin print ('Notify Customer Relations'); Rollback; end

Từ khóa “after” trong câu lệnh trên có nghĩa là gì?

After có nghĩa là sau khi thực hiện hoàn tất lệnh delete, trigger mới được kích hoạt

1. Cho

Create trigger AA on Q for delete as declare @old int select @old=ol.B from deleted ol if (@old >2) Begin print ('Notify Customer Relations');

Từ khóa “for” trong câu lệnh trên có nghĩa là gì?

For có ý nghĩa là với mỗi lần thực hiện một lượt thực hiện delete, trigger sẽ được kích hoạt sau đó.

1. Ý nghĩa của câu lệnh dưới

Create trigger BB on Q1 instead of insert as Declare @newA int, @newB int Select @newA = n.A, @newB = n.B From Inserted n Insert into Q2 values(@newA, @newB)

Tạo trigger BB trên bảng Q1 khi thực hiện lệnh thêm n vào hai bảng A và B. Khai báo newA = n.A, newB = n.B, thêm hai biến trên vào bảng Q2.

1. View là gì?

Trong SQL Server, View là đoạn lệnh truy vấn đã được viết sẵn và lưu bên trong cơ sở dữ liệu. Một View thì bao gồm 1 câu lệnh SELECT và khi chạy View thì sẽ nhận kết quả giống như khi mở 1 Table. Table trên được gọi là một Table ảo. Bởi vì nó có thể tổng hợp dữ liệu từ nhiều Table để tạo thành 1 Table ảo.

1. Mục đích của sử dụng view

Để tập trung, đơn giản hóa và dễ tùy chỉnh tùy theo tùy chỉnh của người dùng dữ liệu

An toàn trong vấn đề bảo mật khi người dùng chỉ có thể truy cập dữ liệu thông qua view mà không phải trực tiếp truy vấn vào dữ liệu gốc

Để cung cấp giao diện mô phỏng bảng mà lược đồ bảng đó có thay đổi

1. Trong SQL server, cú pháp câu lệnh tạo, xóa, sửa view?

*Tạo form:*

CREATE VIEW view\_name AS

SELECT col\_name(s)

FROM table\_name(s)

WHERE condition

*Xóa form:*

DROP VIEW view\_name

*Sửa view:*

ALTER VIEW view\_name AS

SELECT col\_name(s)

FROM table\_name(s)

Where condition

1. Phân loại View?

Updatable View và Un–updatable view

1. Ý nghĩa của câu lệnh

Create view cc

as Select A, max(B) as ln

from Q1

group by A;

Chọn quan hệ Q1 nhóm các bộ theo thuộc tính A sau đó chọn ra cột A và giá trị lớn nhất của cột B trong nhóm A để đổ dữ liệu vào view cc. View cc sẽ bao gồm 2 cột A và cột ln. Cột A sẽ gồm các bộ có thuộc tính A khác nhau. Cột ln chứa giá trị lớn nhất của cột B trong từng nhóm của cột A.

1. Cho PBan(MaPB int, TenPB char(20), MaNQL int). Viết câu lệnh thêm ràng buộc đảm bảo mỗi phòng ban có 1 người quản lý.

ALTER TABLE Pban ADD CONSTRAINT QL\_not\_null NOT NULL(MaNQL)

1. Cho NV(MaNV int, Hoten char(30), Luong int) và câu lệnh alter table NV add constraint chk\_luong\_min check (salary > 10000). Giải thích ý nghĩa của câu lệnh này.

Thêm vào bảng NV ràng buộc chk\_luong\_min kiểm tra nếu lương của NV trên 10000 thì mới có thể tạo dữ liệu nếu không thì dữ liệu sẽ không được tạo

1. Cho NV(MaNV int, Hoten char(30), Luong int, MaPB int) và view được định nghĩa như sau Create view V\_NV(MaPB, SoNV, TongLg, LgTB)

as Select MaPB, Count(\*), Sum(Luong), Avg(Luong)

From NV

Group by MaPB;

Các câu SQL nào thực hiện được trên view V\_NV

1. Select \* From V\_NV
2. Delete from V\_NV Where MaPB = 1 (Do view đang ở dạng un-updatable, để có thể thực hiện lệnh thì phải có trigger để xóa từng dữ liệu các bộ của nó có MaPB = 1)
3. Cả (1) và (2) đều đúng
4. Cả (1) và (2) đều sai

# Assignment 2

Câu 1 Ràng buộc khóa ngoại là gì? Tại sao các ràng buôc như vậy là quan trọng? Tính toàn vẹn tham chiếu là gì?

Ràng buộc khóa ngoại trong bảng cơ sở dữ liệu là trường phù hợp với trường khóa chính trong bảng khác. Trường khóa ngoại được sử dụng để tạo ra mối quan hệ với trường khóa chính của bảng chính. Điều này ngăn cản bạn chèn thêm bất kỳ dữ liệu không hợp lệ trong trường khóa ngoại vì trường khóa ngoại chỉ chấp nhận những giá trị đã có trong trường khóa chính.

Tính toàn vẹn tham chiếu là sự nhất quán về CSDL giữa 2 quan hệ tham chiếu với nhau.

Câu 2. Cho lược đồ CSDL gồm các lược đồ quan hệ sau:



1. Liệt kê tất cả các ràng buộc khóa ngoại giữa các quan hệ này.

+ Thuộc tính cid thuộc bảng Enrolled là khoá ngoại tham chiếu tới bảng courses

+ Thuộc tính cid thuộc bảng Teaches là khoá ngoại tham chiếu tới bảng courses

1. Cho một ví dụ về một ràng buộc (hợp lý) bao gồm một hoặc nhiều quan hệ trên nhưng không phải là ràng buộc khóa chính hay khóa ngoại.

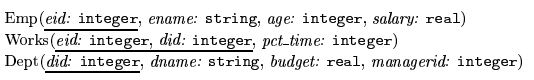
Ví dụ về ràng buộc về gpa :

+ Ràng buộc giá trị từ 0 -> 4 (Với thang điểm 4)

+ Ràng buộc giá trị từ 0 -> 10 (Với thang điểm 10)

1. Viết các câu lệnh tạo CSDL trên bao gồm tất cả ràng buộc có thể có.

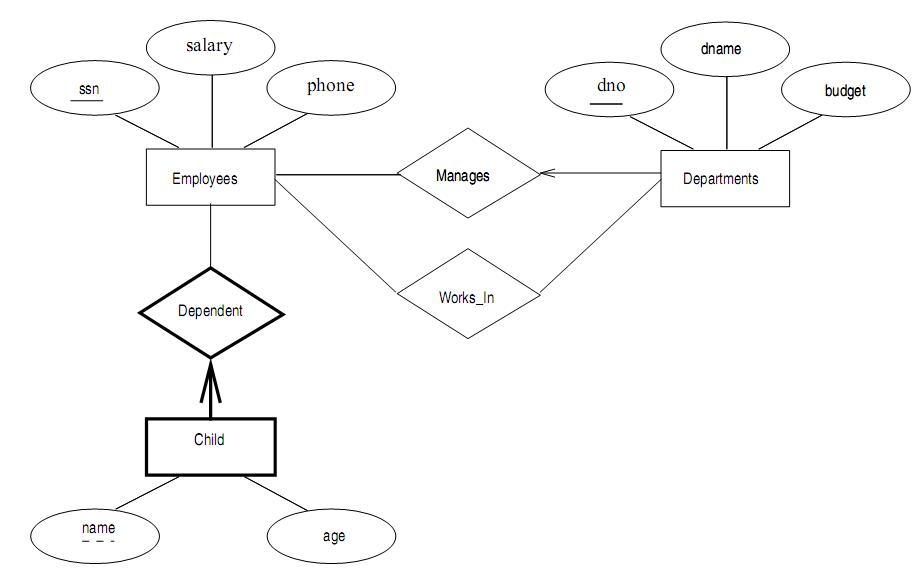
Câu 3. Cho lược đồ CSDL gồm các lược đồ quan hệ sau:



Hãy trả lời các câu hỏi sau :

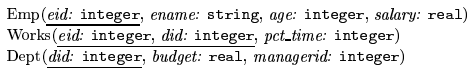
1. Cho một ví dụ về ràng buộc khóa ngoại có liên quan tới quan hệ Dept. Các lựa chọn nào có thể thực hiện để tuân thủ ràng buôc này khi người dùng xóa một bộ trong quan hệ Dept.
2. Viết các câu lệnh SQL để tạo các quan hệ trên bao gồm tất cả các RBTV khóa chính và khóa ngoại.
3. Dùng SQL định nghĩa lại quan hệ Dept sao cho mọi department được đảm bảo có một người quản lý.
4. Viết một câu lệnh SQL để thêm nhân viên ‘John Doe’ với eid = 101, age = 32, và salary = 15.000
5. Viết một câu lệnh SQL để tăng 10% lương cho mọi nhân viên.
6. Viết câu lệnh SQL để xóa department ‘Toy’. Với ràng buộc toàn vẹn tham chiếu bạn đã chọn cho lược đồ này, hãy giải thích điều gì xảy ra khi câu lệnh xóa này được thực hiện.

Câu 4. Giả sử bạn đã thiết kế lược đồ ER cho CSDL một công ty như sau. Hãy viết các câu lệnh SQL để tạo các quan hệ tương ứng bao gồm nhiều nhất các ràng buộc có thể. Nếu có ràng buộc nào bạn không định nghĩa được, hãy giải thích tại sao.



Câu 5.

Cho lược đồ quan hệ sau:



Hãy trả lời các câu hỏi sau.

1. Định nghĩa một ràng buộc mức bảng trên Emp sao cho đảm bảo mọi nhân viên đều có lương ít nhất là 10.000

CREATE TABLE Emp(

eid INT PRIMARY KEY,

ename VARCHAR(255),

age INT ,

salary REAL

CONSTRAINT MinSalary

CHECK (salary >= 1000)

);

1. Định nghĩa một ràng buộc mức bảng trên Dept sao cho đảm bảo tất cả người quản lý đều có tuổi lớn hơn 30.

CREATE TABLE Dept (

did INTEGER,

buget REAL,

managerid INTEGER ,

PRIMARY KEY (did),

FOREIGN KEY (managerid) REFERENCES Emp NOT NULL,

CHECK ( ( SELECT E.age FROM Emp E, Dept D)

WHERE E.eid = D.managerid ) > 30)

1. Định nghĩa một assertion trên Dept sao cho đảm bảo là mọi người quản lý đều có tuổi lớn hơn 30. So sánh assertion này với ràng buộc tạo ở câu 2. Hãy giải thích cái nào tốt hơn.

CREATE TRIGGER CheckAge ON Dept

FOR INSERT AS IF (SELECT Emp.age FROM inserted , Emp WHERE Emp.eid = inserted.managerId ) <= 30

BEGIN

PRINT 'Quan ly phai co tuoi lon hon 30 .'

ROLLBACK TRANSACTION

END

1. Viết câu lệnh SQL để xóa tất cả thông tin về những Employees có lương cao hơn lương của người quản lý của họ. Phải đảm bảo là tất cả các ràng buộc toàn vẹn liên quan phải được thỏa mãn sau khi cập nhật.

Câu truy vấn :

DELETE FROM Emp WHERE (SELECT Salary FROM Emp E, Dept D WHERE D.managerId = E.eid ) < Emp.eid

Câu 6. Hãy cho biết điểm mạnh và điểm yếu của cơ chế trigger. Trình bày sự tương phản giữa trigger với các ràng buộc toàn vẹn khác được hỗ trợ bởi SQL.

Điểm mạnh của Trigger:

Tính toán cập nhật dữ liệu tự động

Kiểm tra dữ liệu nhập

Kiểm tra ràng buộc phức tạp

Bẫy lỗi dể hiểu

Điểm yếu của trigger:

Không tạo và tham chiếu bảng

Không tạo,thay đổi xóa cấu trúc đối tượng sẵn có trong CSDL

Không gán cấp quyền cho người dùng.

Trigger có thể tạo ra 1 bảng ảo mà khi thao tác trên CSDL nó sẽ tự động cập nhật

Câu 7. Cho lược đồ quan hệ dưới đây. Một nhân viên có thể làm viêc ờ nhiều phòng ban. Cột pct\_time trong quan hê Works chỉ tỉ lệ % thời gian nhân viên đó làm việc cho phòng ban tương ứng.

Viết các RBTV, assertions hoặc triggers để đảm bảo các yêu cầu sau:

A. Mỗi nhân viên phải có lương tối thiểu là 1000

Alter table Employee

add constraint check\_salary

check

(

Salary >=1000

)

B. Mọi người quản lý cũng là một nhân viên

create trigger quanLyAsNhanVien01 on Employee

After insert, update, DELETE AS

BEGIN

DECLARE @ketqua INT = (Select count(\*) as soLuong from

(select Super\_ssn as Ssn from Employee

EXCEPT

select Ssn from Employee) as A

where A.Ssn is not null)

if @ketqua>0

begin

rollback;

end

END

c.Tổng tỉ lệ % thời gian làm việc cho các phòng ban của một nhân viên phải dưới 100%.

create trigger fullTimeWorker on Works\_on

after insert, update, DELETE AS

BEGIN

DECLARE @ketqua INT = (SELECT min(soLuongPno)as minSoLuongPno

from

(Select Essn , count(Pno) as soLuongPno

from

(SELECT Essn, Pno

from Works\_on

where [Hours] is not null and Hours >0

GROUP by Essn,Pno) as A

GROUP by Essn) as B)

if @ketqua<2

begin

rollback;

end

END

Một người quản lý phải luôn có lương cao hơn bất kỳ một nhân viên nào mà người đó quản lý.

create trigger CheckSalary on Employee

after insert, DELETE, UPDATE AS

BEGIN

DECLARE @ketqua int = (Select count(\*)

FROM

(SELECT A.Super\_Ssn, B.Salary, max(A.salary)as maxSalarySsn

FROM Employee as A

inner join Employee as B on A.Super\_Ssn=B.Ssn

group by A.Super\_Ssn, B.Salary

having B.Salary <= max(A.Salary))as C)

if @ketqua >0

ROLLBACK;

END

Bất cứ khi nào một nhân viên được tăng lương, lương người quản lý cũng phải được tăng tương ứng.

CREATE TRIGGER Cau7 ON Employee

AFTER UPDATE

AS

BEGIN

UPDATE Employee

SET Salary = Salary + Delta

FROM Employee AS A

INNER JOIN

(

SELECT Sum(ins.Salary - del.Salary) AS Delta, ins.Super\_Ssn AS Super\_Ssn

FROM inserted AS ins

INNER JOIN deleted AS del

ON ins.Ssn = del.Ssn

WHERE ins.Salary > del.Salary

GROUP BY ins.Super\_ssn

) AS B

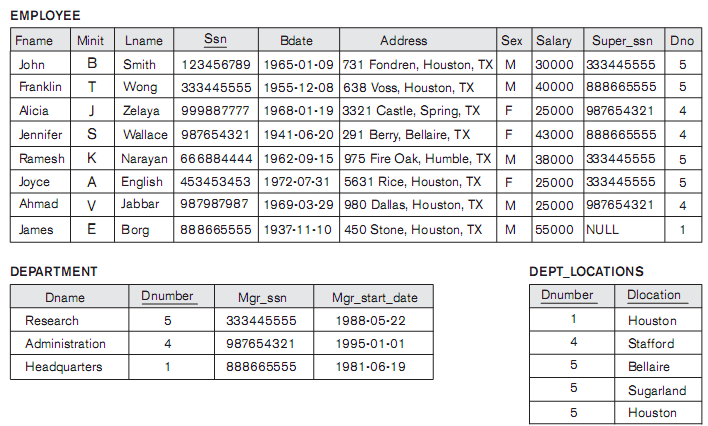
ON A.Ssn = B.Super\_Ssn

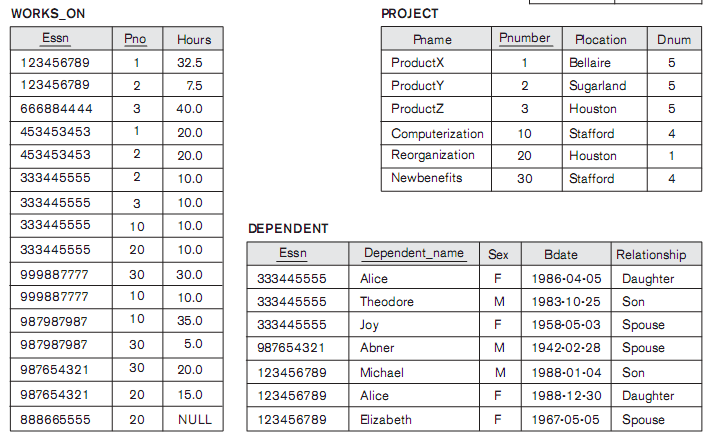
SELECT \* from inserted

END

GO

Bất cứ khi nào một nhân viên được tăng lương, lương người quản lý cũng phải được tăng tương ứng. Hơn nữa, bất cứ khi nào một nhân viên được tăng lương, ngân sách của phòng ban tương ứng cũng phải được tăng lớn hơn tổng lương của tất cả nhân viên thuộc phòng đó.Câu 8. Cho CSDL quan hệ của một công ty như dưới đây.





Dùng SQL để xác định các views sau :

1. Một view có department name, manager name và manager salary của mọi phòng ban.

CREATE VIEW DEPT\_INFO AS

SELECT d.DName , e.Fname , e.LastName, e.Salary

FROM Department d , Employee e

WHERE d.Mgr\_ssn = e.Ssn

1. Một view có employee name, supervisor name và employee salary của mỗi nhân viên thuộc phòng ‘Research’

CREATE VIEW DEPT\_RESEARCH AS

SELECT e.Fname , e.LastName, e.Salary , sp.Fname , sp.LastName

FROM Employee sp , Employee e

WHERE e.Super\_ssn = sp.Ssn

1. Một view có project name, controlling department name, number of employees và tổng số giờ được làm việc mỗi tuần của mỗi dự án.

CREATE VIEW PROJECT\_INFO

AS SELECT p.Pname, d.Dname, p.Pnumber, SUM(w.hours)

FROM Project p, Works\_on w , Department d

WHERE p.Dnum = d.Dnumber AND p.Pnumber = w.Pno

1. Một view có project name, controlling department name, number of employees và tổng số giờ được làm việc mỗi tuần của mỗi dự án với những dự án có nhiều hơn một nhân viên làm việc cho dự án đó.

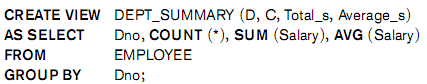
CREATE VIEW PROJECT\_INFO

AS SELECT p.Pname, d.Dname, p.Pnumber, SUM(w.hours)

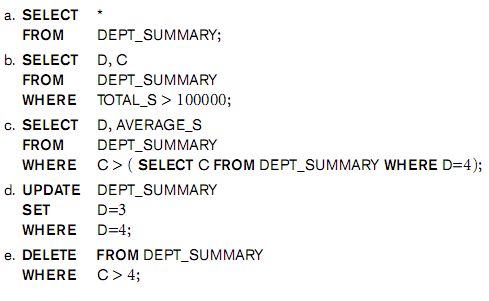
FROM Project p, Works\_on w , Department d

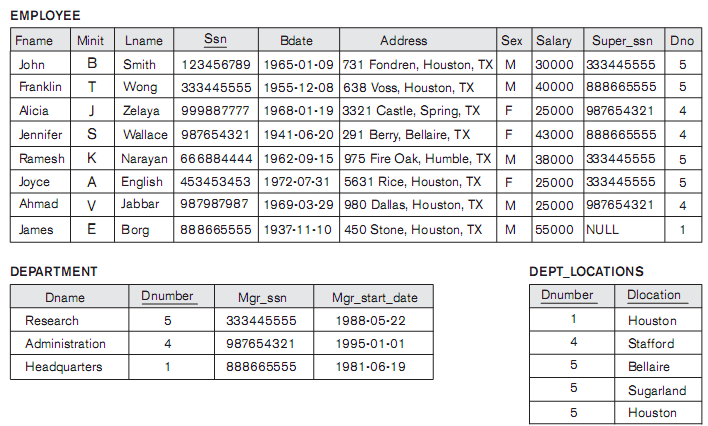
WHERE p.Dnum = d.Dnumber AND p.Pnumber = w.Pno AND p.Pnumber > 1

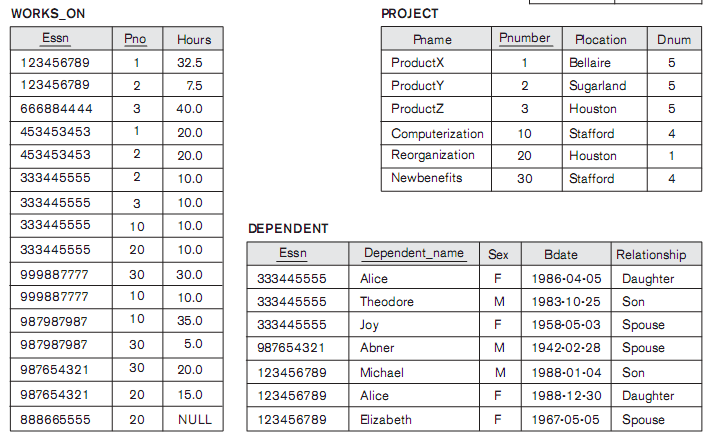
Câu 9 Cho view DEPT\_SUMMARY được định nghĩa trên CSDL một công ty như sau.



Hãy cho biết các truy vấn và thao tác cập nhật nào sau đây được phép trên view. Nếu một query hoặc update nào được phép, hãy cho biết câu truy vấn hoặc cập nhật tương ứng được thực hiện trên các quan hệ cơ sở tương ứng với view.







1. Cho phép . Câu truy vấn có kết quả là :

|  |  |  |  |
| --- | --- | --- | --- |
| D | C | Total\_s | Average\_s |
| 1 | 1 | 55000 | 55000 |
| 4 | 3 | 93000 | 31000 |
| 5 | 4 | 133000 | 33250 |

B. Cho phép. Câu truy vấn có kết quả là

|  |  |
| --- | --- |
| D | C |
| 5 | 4 |

C. Cho phép . Câu truy vấn có kết quả là :

|  |  |
| --- | --- |
| D | Average\_s |
| 5 | 33250 |

D, E : Không cho phép . Chúng ta không thể thay đổi dữ liệu trực tiếp thông qua view . Nhưng chúng ta có thể sử dụng Trigger for view để thay đổi .